

프레임 정보 전처리를 활용한 최적 영상 편집 방법

이준표*, 조철영*, 이종순*, 김태영*, 권철희*

An Optimal Video Editing Method using Frame Information Pre-Processing

Jun Pyo Lee*, Chul Young Cho*, Jong Soon Lee*, Tae Yeong Kim*, Cheol Hee Kwon*

요약

본 논문에서는 편집하고자 하는 위치를 포함하는 GOP(Group Of Picture) 구간만을 선택하여 복호화하고 부호화하는 과정을 수행함으로써 보다 빠르고 정확한 프레임 단위 편집 방법을 제안한다. 제안하는 방법은 편집정보 추출과정 후에 원하는 부분만을 정확하게 분할해 내는 방법과 같은 해상도를 가지는 서로 다른 MPEG 스트림을 병합시키거나 하나의 파일에서 서로 다른 시간을 가지는 부분들을 선택하여 병합시키는 방법으로 나누어 처리한다. 분할은 선택한 프레임이 존재하는 GOP를 복호화 시켜 디스플레이가 정확하게 이루어지도록 필요한 프레임을 포함 하거나 제거한 후 다시 부호화한다. 또한 분할이나 병합 후에 완전한 하나의 스트림으로 구성하기 위해 시각 관리정보의 수정뿐만 아니라 시스템 표준에서 정의하는 구성에 맞도록 전체를 재구성한다. 실험 결과 본 논문이 제안하는 편집방법이 모든 프레임에 걸쳐 정확하고 빠른 분할과 병합 과정을 처리하는 것을 나타낸다.

Abstract

We can cut and paste portions of MPEG coded bitstream efficiently to rearrange the audio and video sequences using our proposed method. The proposed method decodes the MPEG stream within just only one GOP(Group of Picture), edits the decoded video frames, and encodes it back to a MPEG stream. In this method, precise editing is possible. A pre-processing step is specially designed to provide easy cut and paste processing. In the pre-processing step for editing MPEG streams, the detail information is extracted. In addition, video quality is not degraded after the proposed editing process is applied. Consequently, the experimental results show significant improvements compared with traditional algorithms for video editing method in terms of the efficiency and exactness.

▶ Keyword : 영상 편집(video editing), 프레임 정보(frame information), 편집 알고리즘(editing algorithm), 영상 압축 표준(video compression standard)

• 제1저자, 교신저자 : 이준표
• 투고일 : 2010. 05. 06, 심사일 : 2010. 05. 18, 게재확정일 : 2010. 05. 25.
* LIG 넥스원 지휘통제연구센터

1. 서론

MPEG은 90년대에서 21세기에 걸쳐 통신, 방송, 저장 미디어, 컴퓨터 분야에서 공통으로 사용하기 위한 디지털 동영상 부호화, 음향 부호화 그리고 다중 분리방식에 관한 국제 표준이다[1]. MPEG-1 표준화 활동은 MPEG-1 비디오, MPEG-1 오디오 및 MPEG-1 시스템에 대한 활동으로 구성되어 있다. 동화상 정보의 부호화에 대한 표준화를 담당하는 MPEG은 1988년에 ISO/TC97/WG9에서 처음으로 시작되었다. 정지화상 압축 부호화 표준인 JPEG에 비해 MPEG은 동화상 정보를 부호화 하는 것으로서 동화상의 프레임과 프레임 사이에 존재하는 정보의 중복성을 줄여서 보다 높은 압축률을 얻도록 하고 있다. MPEG 비디오 압축 알고리즘은 두 가지 기술을 바탕으로 한다. 시간상의 중복성을 줄이기 위해 블록 단위의 움직임을 보상하고 공간상의 중복성을 줄이기 위해 DCT (Discrete Cosine Transform)에 기반을 둔 압축 알고리즘을 사용한다[2][3]. 즉, MPEG 비디오의 핵심기술은 이전 프레임과 현재의 프레임의 차를 이용하여 움직임을 추정하고 이를 보상해주는 ME/MC (Motion Estimation/ Motion Compensation) 기법과 유효 데이터를 최소화하기 위한 변환 부호화인 DCT의 적용이라고 볼 수 있다. MPEG-1은 H.261에 이어 표준화 작업이 진행되었기 때문에 H.261의 부호화를 거의 이어받은 것이다. 그러나 MC(Motion Compensation)와 DCT(Discrete Cosine Transform)라는 부호화의 기본적인 구조는 같지만 섬세한 부분에서는 몇몇의 차이가 있다. 기본적으로 H.261은 통신매체를 대상으로 하지만 MPEG-1은 저장매체를 위한 것이다. CD 기술이 저장매체로 사용될 수 있게 됨에 따라 이를 고려하여 MPEG-1은 CD-ROM 액세스 속도인 1.2Mbits/s의 전송률을 사용할 수 있도록 데이터를 압축한다. 이렇게 압축을 목적으로 표준화되었기 때문에 동영상의 중간을 임의로 자르거나 잘려진 동영상들의 병합 등은 여러 가지 어려운 점이 따른다. 지금까지는 이러한 편집상의 어려운 점을 해결하기 위하여 전체 MPEG 스트림을 복호화시켜 편집하거나 또는 GOP(Group of Pictures) 단위로의 편집을 수행한다[4]-[8]. 그러나 이와 같은 방법들은 원하는 위치에서 정확한 편집이 불가능하거나 많은 시간을 필요로 하는 문제점을 가지고 있다. 본 논문에서는 이러한 문제점을 해결하여 빠르고 정확한 편집이 가능한 방법을 제시한다.

본 논문에서는 최적의 시간으로 원하는 부분을 분할하거나 병합하고, 편집 과정 후에 손실 없는 디스플레이를 보장하기

위해 시스템, 비디오, 오디오의 세 가지 수준에서 처리한다. 여기서 MPEG 시스템은 여러 개의 압축된 오디오 및 비디오 신호의 다중화 및 동기문제를 다룬다. 동기화를 가능하게 하는 것은 그림 1에서 보는 것과 같이 오디오와 비디오의 packet header에 있는 PTS(Presentation Time Stamp)와 DTS(Decoding Time Stamp)이다. 이중 PTS는 각 액세스 유닛의 재생시간을 의미하며 기준시계가 PTS에 의해 지정된 시각과 일치하면 복호한 오디오와 비디오의 액세스 유닛을 출력함으로써 서로간의 동기화를 이룬다.

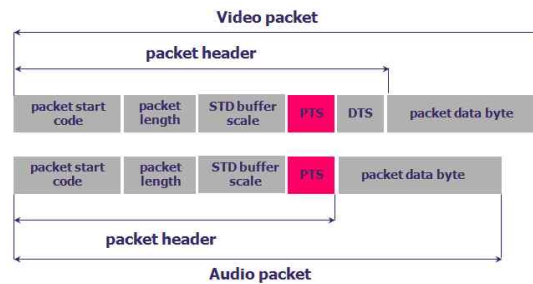


그림 1. 비디오와 오디오의 패킷 구조
Fig 1. Video and audio packet structure

영상 데이터는 시퀀스 층, GOP 층, 픽처 층, 슬라이스 층, 매크로 블록층, 블록 층의 6개의 계층구조로 구성되어 있다. 영상 데이터 전체의 개시를 의미하는 시퀀스 개시 코드와 같이 시퀀스, GOP, 픽처, 슬라이스 등의 각 계층에는 32비트의 바이트 배치된 개시코드가 준비되어 있는데, 이들 개시코드의 비트 패턴은 MPEG-1 비트열 가운데 정해진 위치 이외에는 결코 발생하지 않는다. 시퀀스 오류코드 이외의 개시코드는 신택스 상에서 정의된다. 개시 코드 앞에는 바이트 배치를 유지하기 위해 임의의 개수의 0이 붙는다. 이것은 부호발생량이 예정보다 적을 때 언더플로우를 막기 위해서 더미 비트로서 삽입된다. 또한 MPEG-1 오디오 비트열은 연속적인 프레임들로 구성되며 한 프레임을 AAU(Audio Access Unit)라고 부른다. 그림 2는 MPEG 오디오 비트열이 AAU로 구성되어 있는 상태를 보인다.

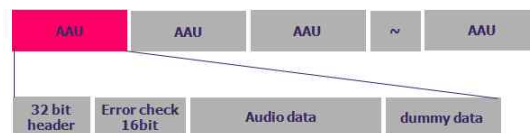


그림 2. 오디오 프레임의 구조
Fig 2. Audio frame structure

AAU는 각각을 단독으로 복호할 수 있는 최소 단위로서 항상 일정한 샘플수의 데이터를 압축하여 싣고 있다. 따라서 하나의 AAU의 비트 수는 평균 384x 전송속도를 샘플링 주파수로 나눈 값이 된다. AAU의 경계는 항상 4바이트 경계에 맞추어지고 상기의 평균 비트 수에서 구할 수 있는 AAU환경이 4바이트 경계에 맞추어지고 다음의 4바이트 경계에 합치될 수 있다. AAU의 내용은 헤더, 옵션 사항인 오류체크, 오디오 데이터 등으로 이어진다. 여기까지가 오디오 신호를 재생하기 위해 쓰이는 데이터이다. 오디오 데이터는 가변 길이 데이터인데 오디오 데이터의 끝이 AAU의 끝에 달하지 않는 경우 남은 부분을 부가데이터(dummy data)라고 부른다. 이 부분에는 MPEG 오디오 이외의 임의의 데이터가 삽입될 수 있다.

본 논문은 다음과 같이 구성된다. 2장에서는 본 논문이 제안하는 효율적인 MPEG 분할 및 병합 방법에 대해 설명하고 3장에서는 본 논문에서 제안하는 방법으로 실험한 결과를 보여주고 마지막으로 4장에서 결론을 맺는다.

II. 제안하는 편집 방법

본 논문에서는 전체 MPEG 파일을 복호화하고 부호화하는 과정을 거쳐 편집하거나 GOP 단위로만 편집하는 기존 방법과 달리 최소의 GOP 구간 내에서만 수행되는 복호화 및 부호화로 최적의 시간으로 모든 프레임들에 대한 정확한 프레임 단위 편집이 가능하도록 하였으며 동시에 손실 없는 디스플레이를 보장할 수 있도록 한다.

1. 편집 정보 추출

제안하는 방법은 MPEG 스트림을 분석하는 단계인 편집정보 추출 과정을 거쳐 원하는 부분만을 정확하게 분할해 내는 방법 그리고 서로 다른 MPEG 파일을 병합시키거나 하나의 파일에서 서로 다른 시간을 가지는 부분들을 선택하여 병합시키는 방법으로 나누어 처리한다. 본 논문에서는 효율적인 편집을 위해 주요한 편집정보를 추출하는 단계를 거친다. 편집정보 추출 과정은 MPEG에 존재하는 필요한 모든 중요 정보들을 요약하여 보관함으로써 분할이나 병합과정을 수행하는데 유용한 정보로서 이용할 수 있도록 하는 단계이다. 그림 3은 편집정보 추출 과정을 수행한 후에 요약된 정보를 나타낸다.

편집정보 처리 과정은 분할과 병합을 위해 필요한 각각의 헤더정보의 위치뿐만 아니라 파일의 길이 및 크기, 해상도, 프레임율 등을 분석, 전체 MPEG 스트림의 세부 정보로 환

용함으로써 정확한 편집이 최적의 시간으로 이루어지도록 하는 역할을 한다. 특히 존재하는 해상도 정보는 서로 다른 부분을 선택하여 하나의 MPEG 스트림으로 만드는 병합과정에 있어 그 가능 여부를 판단하는 중요한 정보로서 활용한다. MPEG 스트림이 가지고 있는 해상도 정보를 얻기 위해 비디오 시퀀스 헤더 정보를 찾아 각각 12bit로 구성되어 있는 horizontal size와 vertical size를 이용하며 프레임 율을 구하기 위해 4bit 의 Picture rate 필드를 이용한다.

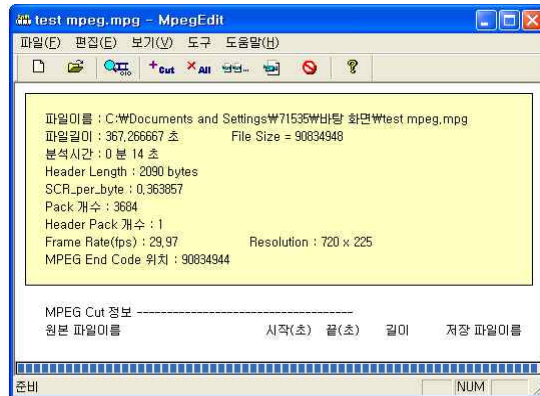


그림 3. 오디오 프레임의 구조
Fig 3. Audio frame structure

2. 분할

본 논문에서는 편집정보 추출 과정 후 최적의 시간으로 분할하고, 분할 과정 후에 손실 없는 디스플레이를 보장하기 위해 비디오, 오디오, 시스템의 세 가지 수준에서 처리한다. 그림 4는 제안하는 MPEG 분할 방법의 전체 구성도이다.



그림 4. 제안하는 MPEG 분할 방법
Fig 4. The proposed MPEG cut method

MPEG은 I, P, B-프레임의 세 가지 타입을 규정한다. 각 프레임들의 서로 다른 특징으로 인해 분할하고자 하는 선택 부분이 어떤 프레임인지에 따라서 처리되는 방법에 차이가 존재한다. 또한 분할과정이 수행될 앞부분의 GOP 처리와 뒷부분의 GOP 처리 역시 서로 다른 방법으로 다룬다. 그림 5는 분할과정이 수행되어야 할 시작부분과 끝 부분의 GOP를 보여준다.

그림 5는 I-프레임을 지정하여 분할하고자 할 때 각 프레임의 순서와 선택된 프레임이 P-프레임일 경우의 각 프레임의 순서 그리고 선택된 프레임이 B-프레임일 경우 각 프레임의 순서를 보인다.

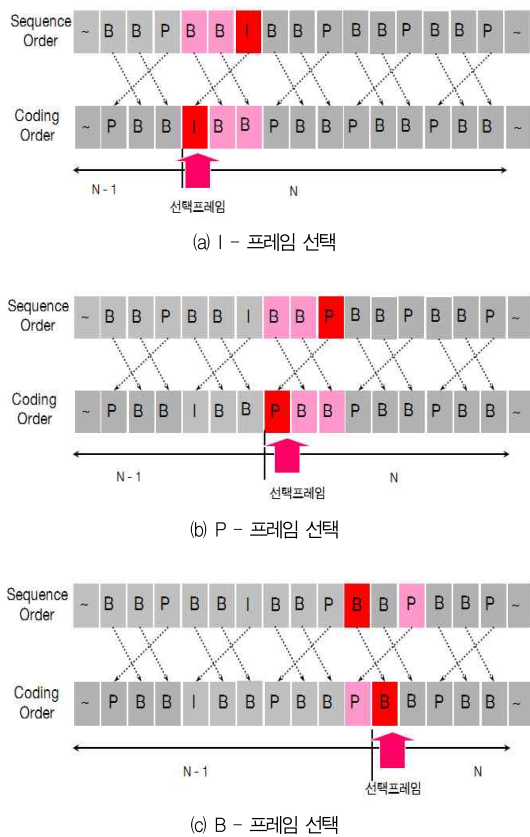


그림 5. 프레임 타입에 따른 분할방법
Fig 5. Out method according to the frame type

본 논문에서 비디오와 동기화 되는 오디오만을 선택하기 위해 Packet 내에 존재하는 33bits로 구성되는 PTS를 참조한다. PTS는 각 액세스 유닛의 재생시간을 의미하며 기준시계가 PTS에 의해 지정된 시각과 일치하면 복호한 액세스 유

닛을 출력하도록 되어있다. 따라서 선택된 비디오의 Packet 내부에 존재하는 PTS 값과 가장 근사한 PTS 값을 가지는 오디오 부분을 선택하여 비디오와 오디오 간의 동기화를 유지한다. SCR(System Clock Reference)은 비디오와 오디오의 복호기를 포함한 MPEG 시스템 복기에 있어서 시각 기준이 되는 System Timing Clock 이라는 기준이 되는 동기정보의 값을 부호기 측에서 의도한 값으로 설정하기 위한 정보이다. 따라서 기준에 타임스탬프는 분할과정을 거침에 따라 정확한 재생 출력을 위해 새롭게 수정되어야 할 필요가 생김에 따라 Pack과 Packet에 존재하는 시간정보를 새롭게 재구성한다. 재구성을 하는데 있어 Pack Layer에서 수정한 필드는 시스템 시각 기준 참조인 SCR이다. SCR은 인코더에서 사용한 시스템 클럭과 디코더에서 사용하는 클럭과 서로 동기를 맞추기 위해 사용되며 식 (1)과 같이 계산한다.

$$SCR(i) = NINT(system\ clock\ frequency \times time(i)) \% 2^{33} \dots (1)$$

여기서, i 는 packet의 시작부분에서 SCR의 마지막 byte까지의 byte count이며 $time(i)$ 는 해당 byte가 system target decoder로 진입될 때의 시간이다.

PTS는 MPEG 시스템의 기준 복호기 내부의 기본이 되는 동기신호와 일치할 때 그 액세스 유닛을 재생 출력하며 33bit로 기록되어있다. 이 필드 역시 분할 과정에서 변화됨에 따라 PTS를 식 (2)를 이용해 새롭게 구성한다.

$$PTS = (PTS + PTS\ difference) \% 2^{33} \dots (2)$$

여기서, $PTS\ difference$ 는 현재 PTS 가 가지고 있는 값과 처음 나타나는 PTS 가 가지고 있는 값의 차이이다.

비디오의 부호화 비트열의 송출 순서가 특별하기 때문에 복호화 된 시간을 기록하기 위해 필요한 필드이며 33bit로 기록된다. DTS는 식 (3)을 이용하여 계산한다.

$$DTS = (DTS + PTS\ difference) \% 2^{33} \dots (3)$$

3. 병합

같은 해상도를 가지는 서로 다른 MPEG 스트림이나 하나의 스트림에서 서로 다른 시간을 가지는 부분들을 선택하여 병합과정을 수행한다. 그림 6은 제안하는 MPEG 병합 방법의 전체 구성도이다.



그림 6. 제안하는 MPEG 병합 방법
Fig 6. The proposed MPEG paste method

병합을 위해 최소 2개 서로 다른 비디오 시퀀스를 선택하며 선택된 비디오 영역을 분할 과정을 통해 각각 분할한다. 이와 동시에 비디오 시퀀스가 가지고 있는 PTS를 이용하여 그와 같거나 가장 유사한 PTS값을 가지는 오디오 영역을 선택, 분할한다. 이렇게 분할된 비디오와 오디오 영역을 그림 7과 같이 하나의 스트림으로 재구성한다. 병합과정은 완전한 하나의 MPEG 스트림으로 구성해 주는 과정이다. 따라서 위에서 제시한 바와 같이 시각 관리정보의 수정뿐만 아니라 시스템 수준의 표준에서 정의하는 구성에 맞도록 전체를 재구성한다.



그림 7. 비디오 재구성 방법
Fig 7. Video reconstruction method

III. 실험결과

본 논문에서 제안된 편집 방법의 실험을 위해 Pentium-4 1.4G Hz CPU에서 Visual C++로 제작된 프로그램에서 실행한다. 실험 데이터로써 MPEG-1 데이터를 이용하여 수행하며, 평가방법은 기존의 편집방법과 제안하는 편집방법을

활용한 편집 수행 시간을 이용한다. 표 1 은 수행된 실험 결과를 나타낸다.

표 1. 처리시간 결과
Table 1. Processing time

영상의 시간	기존방법		제한하는 방법
	전체복호화 후 부호화	GOP 단위	
10초	1분25초 72	0.49초	10초 40
20초	2분 51초 19	0.67초	10초 53
30초	4분 17초 91	0.81초	11초 53
40초	5분 43초 35	1.04초	11초 88
50초	7분 09초 78	1초 72	14초 53
1분	8분 36초 81	2초 40	16초 02
5분	44분 01초 20	34초 01	47초 97
10분	1시간 31분 58초	1분 21초 55	1분 57초 23

제한하는 방법은 전체 복호화 후 부호화하는 방법에 비해 전체 실험 영상 시간에 걸쳐 상당히 빠른 수행 결과를 보이고 있으나 GOP 단위 편집 방법에 비해서는 비교적 더 많은 시간을 더 필요로 한다. 그러나 이러한 편집 시간의 차이는 분할하고자 하는 영상의 시간이 길어질수록 그 차이는 점차 감소함을 보인다. 즉, 10분의 영상 시간을 가지는 데이터를 대상으로 수행된 실험에서 기존 방법인 GOP 단위가 1분 21초 55에 비해 제안하는 방법이 1분 57초 23으로 수행 시간의 차이가 비교적 적음을 알 수 있다. 이는 현재 사용되고 있는 영상이 60분에서 90분의 실행 시간을 갖는다는 것을 고려하였을 때 제안하는 영상 편집 방법이 기존의 편집 방법에 비해 정확하고 빠른 편집을 수행함으로써 더 효율적이라고 할 수 있다.

IV. 결론

MPEG은 압축을 목적으로 표준화되었기 때문에 동영상의 중간을 임의로 자르거나 잘려진 동영상들의 병합 등은 여러 가지 어려운 점이 따른다. 지금까지는 이러한 편집상의 어려운 점을 해결하기 위하여 전체 MPEG 스트림을 복호화시켜 편집하거나 또는 GOP 단위로의 편집을 수행하였다. 그러나 이와 같은 방법들은 원하는 위치에서 정확한 편집이 불가능하거나 많은 시간을 필요로 하는 문제점을 가지고 있다. 이를 위해 본 논문에서는 MPEG 시퀀스를 효율적으로 편집하는 방법을 제안한다. 특히 MPEG의 특징을 이용해서 최적의 시간으로 정확한 편집이 가능한 방법을 제안하고 있으며 오디오와 비디오와의 동기화를 이루는 방법을 제안한다.

제안하는 방법은 MPEG 스트림을 분석하는 단계인 편집 정보 추출 과정을 거쳐 원하는 부분만을 정확하게 분할해 내는 방법 그리고 서로 다른 MPEG 파일을 병합시키거나 하나의 파일에서 서로 다른 시간을 가지는 부분들을 선택하여 병합시키는 방법으로 나누어 처리한다. 제안하는 방법의 효율성을 보이기 위해 수행된 실험을 통해 제안하는 방법이 분할 및 병합에 있어 기존의 방법에 비해 보다 정확하고 신속하게 수행됨을 보인다.

참고문헌

[1] D. L. Gall, "MPEG: A Video Compression Standard for Multimedia Applications," Proc. of Int. Conf. ACM, Vol.34, No.4, pp. 47-58, April 1991.

[2] 정명범, 김재경, 고일주, 장대식, "특수 영상에서 비디오 요약을 위한 장면 전환 검출 알고리즘," 한국컴퓨터정보학회논문지, 제 14권, 제 3호, 2009년 3월.

[3] 정영범, 고일주, "Low Peak Feature와 영상 Color를 이용한 유사 동영상 검색," 한국컴퓨터정보학회논문지, 제 14권, 제 4호, 51-58쪽, 2009년 8월.

[4] J.Meng and S.-F Chang, "CVEPS: A Compressed Video Editing and Parsing System," Proc. of Int. Conf. ACM, Boston, MA., Nov., 1996.

[5] Jiun Shiu, Shou Chia Cheng, Ja Ling Wu and Chun Hung Lin, "A low cost editor for MPEG-1 system streams," IEEE Trans. on Consumer Electronics, Vol. 41, pp. 620-625, Aug. 1995.

[6] Markov, P, and Mehrpour, "Rate renegotiation algorithm with dynamic prediction window for efficient transport of streaming VBR MPEG coded video over ATM networks," Proc. of Int. Conf. ICON, pp. 83-88, 2002.

[7] Pazarci, M and Dicipin, V, "Data embedding in scrambled digital video," Proc. of IEEE Int. Symp., on Computers and Communication, pp. 498-503, July, 2003.

[8] Ching-Yung Lin, Tseng, B.L, and Smith, J.R., "Universal MPEG content access using compressed-domain system stream editing technique," Proc of IEEE Int. Conf. on Multimedia and Expo, Vol. 2, pp. 73-76, Aug. 2002.

저자 소개



이준표

2008: 한양대학교 공학박사
 2001-2003: 파인드테크(주) 연구원
 2003-2009: 한양대학교 공학기술연구소 연구원
 2009-현재: LIG넥스원(주) 선임연구원
 관심분야: 영상처리, 멀티미디어 시스템, 패턴인식



조철영

2008: 충남대학교 공학사.
 2008-현재: LIG넥스원(주) 연구원
 관심분야: 패턴인식, 지능제어, 인지과학



이종순

1982: 한양대학교 이학박사
 1984: 한양대학교 이학석사
 1986-현재: LIG넥스원(주) 수석연구원 (팀장)
 관심분야: 시스템 제어, 정보 및 영상 신호처리



김태영

1985: 경북대학교 공학사
 1987: 경북대학교 공학석사
 2009: 성균관대학교 공학박사
 1990-현재: LIG넥스원(주) 수석연구원 (센터장)
 관심분야: 디지털신호처리, 컴퓨터응용



권철희

2000: 고려대학교 공학석사
 2000-현재: LIG넥스원(주) 책임연구원
 관심분야: 패턴인식, HCI, 디지털 신호 처리, 컴퓨터 기술 응용